# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



**1. REPORT DATE** *(DD-MM-YYYY)*
25-07-2016

**2. REPORT TYPE**
final

**3. DATES COVERED** *(From - To)*
05/01/2015 - 04/30/2016

**4. TITLE AND SUBTITLE**
Host immunity via mutable virtualized large-scale network containers

**5a. CONTRACT NUMBER**

**5b. GRANT NUMBER**
N00014-15-1-2026

**5c. PROGRAM ELEMENT NUMBER**

**6. AUTHOR(S)**
Sun, Kun

**5d. PROJECT NUMBER**

**5e. TASK NUMBER**

**5f. WORK UNIT NUMBER**

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
The College of William and Mary, Office of Sponsored Programs, P.O. Box 8795, Williamsburg, VA 23187-8795

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Office of Naval Research

875 N Randolph St, Arlington, VA 22217

**10. SPONSOR/MONITOR'S ACRONYM(S)**
ONR

**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

**12. DISTRIBUTION/AVAILABILITY STATEMENT**
Approved for Public Release; Distribution is Unlimited.

**13. SUPPLEMENTARY NOTES**

**14. ABSTRACT**
The relatively static configurations of networks and their hosts allow attackers to gather intelligence, perform planning, and execute attacks at will. We propose a scalable, dynamic, adaptive system for host immunity that combines virtualization, emulation, and mutable network configurations. This system is deployed on a single host, and provides host protection through hiding the real system among a large number of decoys with dynamic virtualized network topology. It will make the network scanner and intruder spend more time and effort on attacking the worthless targets (decoys).

**15. SUBJECT TERMS**
Moving Target Defense, Decoy, Dynamic Virtualized Network

| 16. SECURITY CLASSIFICATION OF: a. REPORT | b. ABSTRACT | c. THIS PAGE | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON / 19b. TELEPHONE NUMBER *(Include area code)* |
|---|---|---|---|---|---|
| U | U | U | UU | 8 | Kun Sun / 7572213457 |

**Standard Form 298** (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

# "Host Immunity via Mutable Virtualized Large-Scale Network Containers"

PROJECT: ONR-BAA-15-001
Contract #: N00014-15-1-2026

Final Technical Report
Reporting Period: 1 May 2015 – 30 April 2016

Principal Investigator: Dr. Kun Sun
Department of Computer Science, P.O. Box 8795
College of William and Mary
Williamsburg, VA 23187-8795
757-221-3457 (voice); 757-221-1717 (fax); ksun@wm.edu

# Contents

# 1 Project Objectives

The relatively static configurations of networks and their hosts allow attackers to gather intelligence, perform planning, and execute attacks at will. This project's overall objective is to protect the real system running on a host computer via mutable virtualized large-scale network containers. Specific objectives include (1) increasing the attack surface and hiding the real system in a large number of decoys, and (2) developing dynamic virtualized network topology on a host computer. If successful, our host immunity system could be used by the Navy to better protect its computer system from the existing attacks and enhance the resilience of the services and applications when it is under attacks. Even if an attacker gains some access to the host, his ability to exploit the penetration is limited because what he obtained is no longer true. As time goes on our system knows more about the attacker while he knows less about our system.

# 2 Summary of Research Activities

There are two major challenges when designing our large-scale dynamic virtualized network system. We perform two major tasks during the non-cost extension period between 05/2015 and 04/2016 after the PI moved to College of William and Mary. First, the system should sustain service availability to the authenticated clients when the system changes its topology. The network connections should be maintained when the IP or Port number is changing. Meanwhile, the service downtime should be minimized. We develop a virtual dynamic network framework, which enables network administrators to dynamically change the network topology and the network configuration based on the attackers' activities. We propose a VM-based seamless TCP connection migration scheme to support live VM migration without losing the existing network connections. We port the VM migration kernel module from Linux kernel version 2.4 to 2.6. Moreover, we propose a novel defense mechanism for protecting the identity of nodes in Mobile Ad Hoc Networks and defeat the attacker's reconnaissance efforts. To preserve communication among legitimate nodes, we modify the network layer by introducing a translation service for mapping virtual identities to real identities, a protocol for propagating updates of a node's virtual identity to all legitimate nodes; and a mechanism for legitimate nodes to securely join the network. We show that the proposed approach is robust to different types of attacks, and also show that the overhead introduced by the update protocol can be controlled by tuning the update frequency.

Second, we need to mitigate the threat from insiders. Though the system only allows authenticated clients to locate and access it service, insiders (as authenticated clients) can still connect to the real system and perform attacks. We develop an anti-crawler mechanism called PathMarker to detect and constrain the distributed persistent inside crawlers that have valid credentials to access the web services. The main idea is to add a marker to each web page URL and use the URL path and user information contained in the marker to help accurately detect crawlers at its earliest stage. PathMarker can dramatically suppress the efficiency of distributed crawlers and effectively reduce the crawling speed of individual persistent crawler.

# 3 Final Goals

This project will develop a set of optimized, validated, and fully documented algorithms and mechanisms. A successful system prototype will be delivered and lead to a powerful new capability for using moving target defense mechanism to build resilient, adaptive, and secure systems. The primary deliverable of this effort will be the actual secure system based on moving target defense mechanisms, for deployment on commodity computers. Our system will be integrated in open-source software system like KVM. The software will be delivered as a series of software drops with incremental capabilities. There are no proprietary claims associated with our deliverables.

# 4 Technical Approaches

1. **Ensuring service availability.**

- **DESIR: Decoy-Enhanced Seamless IP Randomization**

Sophisticated adversaries usually initiate their attacks with a reconnaissance phase to discover exploitable vulnerabilities on the targeted networks and systems. To mitigate the effectiveness of persistent reconnaissance attacks, we develop a defensive mechanism that dynamically mutates network topology with a large number of decoys to invalidate the attacker's knowledge from network scanning. We combine the IP randomization technique with decoy techniques and solve two challenges, namely, service availability to legitimate users and service security against unauthorized users. First, our solution can minimize the probability of the real servers being identified and compromised by unauthorized users through deploying a large number of decoy nodes, which change their IP addresses along with the real servers to prolong the scanning time of the attackers. Second, our solution can ensure seamless connection migration so that all existing communication connections between the legitimate users and the servers are always kept alive even after the servers migrate to different IP addresses multiple times. We implement a virtual machine based system prototype and evaluate it using state-of-the-art scanning techniques. Both theoretical analysis and experimental results show that our solution can effectively mitigate network reconnaissance attacks without sacrificing service availability.

2. **Ensuring service security.**

- **PathMarker: Defending Persistent Malicious Crawlers**

In 2014, bots account for more than half of all website traffic, and malicious bots contributes almost one third of the traffic. As one type of bots, web crawlers have been manipulated to collect valuable website contents without permission from website administrators. It is still a challenge to detect stealthy distributed persistent crawlers, where individual crawler imitates the real user and a large number of crawlers coordinate to speed up the crawling tasks. In this work, we develop an anti-crawler mechanism called PathMarker to detect and constrain the distributed persistent crawlers. The main idea is to add a marker to each web page URL and use the URL path and user information contained in the marker to help accurately detect crawlers at its earliest stage. PathMarker can dramatically suppress the efficiency of distributed crawlers and effectively reduce the crawling speed of individual persistent crawler. We deploy our approach on a forum website, and the evaluation results show that PathMarker can successfully capture all 12 open-source and in-house crawlers, plus two external crawlers (i.e., Googlebots and Yahoo Slurp), in a short time after the crawlers initiate their crawling tasks.

# 5 Advantages of the Approaches

In this section, we will summarize the advantages of the proposed approaches separately.

- **DESIR: Decoy-Enhanced Seamless IP Randomization**

1. We propose a decoy-enhanced seamless network address randomization framework for constructing dynamically mutable networks to thwart persistent reconnaissance attacks against targeted servers.
2. Our solution supports seamless connection migration with network address randomization. It has good scalability to seamlessly migrate a large number of network connections after the servers change their IP addresses multiple times.
3. We implement a VM-based prototype. The experimental results show that the system overhead is small and our system can effectively defeat persistent reconnaissance attacks.

- **PathMarker: Defending Persistent Malicious Crawlers**

1. We develop an anti-crawler mechanism named PathMarker to accurately detect web crawlers. It can differentiate normal users from malicious crawlers using the URL visiting path and URL visiting timing features derived from the URL marker.
2. PathMarker is able to instantly detect distributed crawlers that share links with each other. If the distributed crawlers do not share links in a pool, our encrypted URL technique can effectively suppress their efficiency. We can reduce the download rate of individual persistent crawler to the level of normal human beings.
3. We implement a PathMarker prototype on an online forum website and show that it is simple to deploy our mechanism on existing websites. The experimental results show that PathMarker is capable of detecting a number of state-of-the-art crawlers.Current State

# 6 Current State

## 6.1 FY16 (5/15-4/16) Research Progress

### 6.1.1 DESIR: Decoy-Enhanced Seamless IP Randomization

In advanced persistent threat (APT), well-resourced and trained adversaries typically initiate the attacks with thorough reconnaissance to gather intelligence about the targeted networks and systems. Once one vulnerability is identified, the adversary can proceed to mount customized exploits to compromise the system. This attacking strategy has been working well due to the static nature of the current network configurations.

In recent years, researchers have proposed to mitigate reconnaissance attacks by dynamically shifting the network attack surface including IP and MAC addresses, open ports, and network topology. In general, by proactively changing the host IP addresses and the network topology, the entire network can be made unpredictable so that the vulnerabilities discovered by an attacker in an early stage become obsolete and useless. However, all those IP randomization based solutions face two challenges. First, though the size of available IP address pool is large, due to the small number of alive IP addresses at one time, the attackers may still complete scanning the entire targeted network quickly and compromise the targeted system before the next round of IP randomization. For instance, ZMap is capable of surveying the entire IPv4 address space within

45 minutes from a single machine. Second, when the servers change their IP addresses, existing active connections may be disrupted, since high-layer protocols such as TCP or UDP depend on a stationary IP address. Therefore, it is a challenge to seamlessly migrate all existing network connections to the new IP addresses with a minimal migration time.

We develop a decoy-enhanced seamless network address randomization mechanism called DESIR to defeat network reconnaissance attacks and ensure service availability. First, we fortify the IP randomization technique with a large number of decoys to protect the servers against reconnaissance attacks. Besides the real servers, we deploy a number of decoy nodes that will change their IP addresses along with the real servers. Decoys have been widely used to distract attacker's attention from the real system; however, APT attacks may eventually identify the decoy nodes based on their response time and fingerprint analysis after interacting with the decoys. In our solution, in addition to deploying a large number of decoys, we randomly shuffle the IP address space of the target network including both the real servers and the decoys. Therefore, though the attacker may create a blacklist of decoy IP addresses through reconnaissance, this blacklist becomes invalid after the next round of IP randomization, and the attacker has to start over the reconnaissance process. In other words, we combine both IP randomization technique and decoy technique to effectively defeat persistent reconnaissance attacks, though neither of them can achieve this goal by itself only.

Second, we develop a seamless network connection migration mechanism to keep alive the existing connections between legitimate users and the servers even after the servers change their IP addresses multiple times. The basic idea is to separate the connection's transport identification from its network identification so that the dynamic changes of network addresses are transparent to the transport layer and the application layer. We introduce a pair of internal addresses to identify the transport endpoints and another pair of external addresses to identify the network endpoints. The internal address remains consistent during the life of the connect session and the external address is changed as the server migrates. Moreover, we guarantee that the legitimate users can always locate the servers and initiate service requests by using a trusted authentication server. Whenever a server changes its IP address, it will notify the updated IP address to the authentication server. When a client wants to connect to the real server, it first authenticates itself to the authentication server, which then sends the server's current IP address to the client.

We evaluate the effectiveness of our decoy-enhanced IP randomization mechanism through both theoretical analysis and real prototype implementation. Our theoretical analysis shows that decoy-enhanced IP randomization can effectively prolong the attacker's scanning time. Suppose one real server is protected by $n$ IP addresses, where $n-1$ IP addresses are occupied by decoy nodes. When the attacker is not aware of our defense mechanism, it may only scan the entire IP address space once either sequentially or randomly. In this scenario, our IP randomization technique can increase the average number of probes from 0.5n to 0.63n. When the attacker knows that the real system is protected by our defense system, it may scan the entire IP address space multiple times, and it will increase the average number of probes from $0.5n$ to $n$. More importantly, the attacker has to spend tremendously more time to distinguish the decoys from the real system, and there is high probability that the attacker will be trapped into one decoy instead of the real server.

We implement a virtual machine (VM)-based prototype that integrates decoy-enhanced IP address randomization with seamless connection migration. The experimental results show that

the overheads for both decoy deployment and IP randomization are reasonably low and can defeat the practical scanning attacks using tools such as Nmap or ZMap.

### 6.1.2 Defending Persistent Malicious Crawlers

With the prosperity of Internet, data collection from the network has gained increasing demands. As one type of bots, web crawlers have been leveraged by search engines (e.g., Googlebot by Google) to popularize websites through website indexing. However, the number of malicious bots is increasing too. To regulate the behavior of crawlers, most websites include a file called "robots.txt" that contains the rules on what information cannot be collected by the web crawlers. The file may even set different rules for different crawlers. However, "robots.txt" only provides a guideline, and almost all malicious robots ignore it. Moreover, since this file is publicly available, malicious crawlers may use it to infer the locations of valuable web contents that the servers want to protect.

For websites that require open user sign-up and login to access more website contents (e.g., sourceforge. net, stackoverflow.com), they don't want to see a small number of free user accounts can successfully download all their web contents. For websites that may contain confidential information or paid documents (e.g., www.sciencedirect.com, ieeexplore. ieee.org), though only authorized or paid users are allowed to access the websites, they still need to prevent authorized users as insiders from collecting the entire website. Particularly, it is a challenge to detect and constrain distributed persistent crawlers against those websites. First, since the crawlers are persistent, they can afford to lower the download rate and better mimic the access behaviors of real users. Thus, it is critical to detect a persistent crawler accurately at its earliest stage. Second, a number of users may coordinate and use a divide-and-conquer strategy to speed up the crawling tasks. Defenders should suppress the efficiency of distributed crawlers to minimize the information leakage before the crawlers are detected.

In this work, we develop an anti-crawler mechanism called PathMarker to detect and constrain the stealthy distributed persistent crawler. The main idea is to add a marker to each web page URL and use the web page path and user information contained in the markers to help identify and confine crawlers. Given one website, we can automatically append a marker to each web page's URL, and we call it URL marker. A URL marker records the information about its parent web page's URL and the user ID who collects the URL. Thus, when distributed crawlers share collected links in a pool, we can detect them through a user ID mismatch, since the user who collects the page may not be the same as the one who visits the URLs contained in this page. Moreover, we encrypt the URL marker along with the original URL except the root URL to further protect the website structure against distributed crawlers, who will see different web links about the same web page to different user IDs.

With the aid of URL marker, we can calculate the depth and width of each page and build accurate URL visiting path based on parent URL recorded in the URL marker. Next, we can leverage machine learning techniques to detect crawlers based on the different URL visiting path patterns and URL visiting timings between human beings and malicious crawlers. We adopt Support Vector Machine (SVM) to model the normal users and crawlers using the features related to the URL marker. Moreover, to lower the false positive rate, we rely on CAPTCHAs to differentiate normal users from crawlers.

We develop a PathMarker prototype on an online forum website. Since our solution needs to make small changes on the source code of web servers, it is difficult for us to test on public

large-size or medium size websites. Instead, we setup a forum website from scratch and integrate our anti-crawler mechanism. We first train a SVM model based on the logging data collected from more than 100 normal users and 6 in-house crawlers, and then test the model using 6 open source crawlers and another set of normal user data. The experimental results show that our anti-crawler technique can effectively detect all the crawlers. Moreover, two external crawlers, Google bot and Yahoo Slurp, are also detected.

# 7 Publications

- Jianhua Sun and Kun Sun, DESIR: Decoy-Enhanced Seamless IP Randomization. To appear in IEEE International Conference on Computer Communications (INFOCOM), San Francisco, CA, April 10-15, 2016.
- Shengye Wan, Yue Li, Kun Sun, **PathMarker: Defending Persistent Malicious Crawlers,** under submission to conference 2016.